

Approximate MMSE Estimator for Linear Dynamic Systems with Gaussian Mixture Noise

Leila Pishdad, Fabrice Labeau

Abstract—In this work we propose an approximate Minimum Mean-Square Error (MMSE) filter for linear dynamic systems with Gaussian Mixture noise. The proposed estimator tracks each component of the Gaussian Mixture (GM) posterior with an individual filter and minimizes the trace of the covariance matrix of the bank of filters, as opposed to minimizing the MSE of individual filters in the commonly used Gaussian sum filter (GSF). Hence, the spread of means in the proposed method is smaller than that of GSF which makes it more robust to removing components. Consequently, lower complexity reduction schemes can be used with the proposed filter without losing estimation accuracy and precision. This is supported through simulations on synthetic data as well as experimental data related to an indoor localization system. Additionally, we show that in two limit cases the state estimation provided by our proposed method converges to that of GSF, and we provide simulation results supporting this in other cases.

Index Terms—Bayesian tracking, linear estimation, Gaussian mixture noise, Gaussian sum filter, minimum mean-square-error (MMSE) estimator

I. INTRODUCTION

The problem of estimating the unobservable state of a dynamic system from its available noisy measurements is prevalent in numerous signal processing contexts. Bayesian tracking techniques have been used for this purpose by using a probabilistic framework and approximating the posterior or *belief function*, i.e. the conditional probability density function of the state given the measurements. For the special case of Gaussian noise with linear dynamic and measurement models, this posterior is Gaussian and its sufficient statistics are optimally tracked by a Kalman filter [1], [2]. The mean of this pdf acts as the state estimate and it is proved to be the minimum mean-square error (MMSE) estimator [3]. However, for the case of non-Gaussian noise processes, approximations should be made to provide suboptimal solutions [2].

Gaussian sum approximation has been an attractive method for estimating non-Gaussian distributions, since it provides asymptotically unbiased estimations [4], with the desired precision¹ [3, Chapter8; Lemma 4.1]. Additionally, by using Gaussian mixtures (GM), the approximated pdf is represented as a conditionally Gaussian distribution and this enables the analytic evaluation of a closed-form expression for the belief function. This is possible since, with GM distributions, the *multiple model* approach can be used, where each component in the GM corresponds to a model in the system and can be

tracked by a Kalman filter [5]. Hence, the partitioned posterior can be estimated by a bank of Kalman filters, i.e. the Gaussian sum filter (GSF). Consequently, GMs have been widely used to model the different non-Gaussian distributions in sequential Bayesian tracking, including the prior [3], [6]–[10], likelihood [11], [12], predictive density [13]–[15], and noise distributions [5]–[7], [10], [16]–[22]. They have also been used to directly approximate the posterior distribution [15], [18], [22], [26]–[29], [32].

With GM prior, likelihood, or predictive density, the posterior is also a GM and the number of its components remains constant over time, as long as the noise distributions are Gaussian. For instance, in [3], it is shown that starting with a GM prior², and additive white Gaussian noise, the predictive and posterior distributions will also be GMs with the same number of components. However, for GM noise distributions, the number of models in the system and consequently the number of components in the posterior grow exponentially over time. Hence, suitable Gaussian Mixture reduction algorithms should be used, to merge or remove some of the components in the posterior as time progresses.

The mixture reduction algorithms can be categorized into three classes. In the first group, Expectation maximization (EM) is used to simultaneously predict and reduce the GM [10], [12], [19], e.g. by running the EM algorithm on synthetically generated data [10], [19]. The second class of reduction algorithms rely on merging a pair or a group of components, i.e. replacing them by their moment-matching Gaussian distribution. There are different criteria for selecting the components to be merged. For instance, in Gaussian pseudo Bayesian (GPB) estimators, the components with the same state history are merged and replaced by a single Gaussian distribution [5]. A less computationally complex solution, also approximating a GM by a single Gaussian, is interacting multiple models (IMM) [5] which is used commonly as it requires fewer filters. Alternatively, in [22] the components are merged in the unlikely regions of the distribution, and they are split in the likely regions. In [34], the component with the smallest weight is merged with its closest neighbor.³ Optimization techniques can also be used to select the merging components such that a cost function which quantifies the dissimilarity between the GM distribution and the reduced distribution is minimized. Some of the cost functions used in the literature include Kullback-Leibler divergence (KLD) [18], [35], and integral square difference⁴ (ISD) [36], [37]. Alter-

The authors are with the Department of Electrical and Computer Engineering, McGill University, Montreal, QC H3A 0E9, Canada (e-mail: leila.pishdad@mail.mcgill.ca; fabrice.labeau@mcgill.ca)

¹The integral of the approximation error over the sample space can be made as small as desired.

²with a finite number of components

³The neighboring components have the smallest Euclidean distance between their means.

⁴This criterion is also referred to as integral square error (ISE)

natively in [38] the authors merge the components leading to the minimum increase in the *within-component* variance of the reduced GM distribution. The last category of reduction schemes requires removing a group of components [6], [7]. This class has the lowest computational complexity, especially since the reduction can be done before tracking, making it possible to avoid the evaluation of unused parameters. In the extreme case, the active component of the posterior is determined and all the other components are removed [17], [20]. This is equivalent to making a hard decision about the model in effect and hence the accuracy and precision of estimation is dependent on the correct choice of this model. At the cost of increased computational complexity, the performance of these methods can be improved by applying *resampling* procedures [16], [28], [39]. Alternatively, in [15], [21] a *forgetting and merging* algorithm is proposed, where the components with weights smaller than a given threshold are removed, and the components with close enough moments are merged. A comprehensive review of the reduction algorithms for GM distributions is provided in [40].

The MMSE estimator is the expected value of the posterior [5], [41]. Hence, GSF is the MMSE estimator of state⁵ [19], [42], [43]. However, due to using parallel Kalman filters in GSF, the MSE of each individual filter is minimized irrespective of the location of its mean with respect to the other components. Yet, the total covariance matrix of the filter is a function of both the individual filters' state estimation covariance matrices and the spread of their means (as we show in (21)–(22) in Section III).

In this work, we propose an approximate MMSE state estimator called AMMSE, which, unlike GSF, minimizes the trace of the total covariance matrix of the filter rather than the traces of the covariance matrices of individual filters. For this purpose, we re-derive the gains of individual filters such that the trace of the total covariance matrix of the filter is minimized. Hence, the spread of the means of the posterior components in the proposed estimator is smaller than that of GSF, making it more robust to removing some of the components. Consequently, AMMSE estimator can be used with the simplest and least computationally complex reduction scheme and yet achieve better performance when compared with GSF with the same reduction scheme. In other words, rather than using more computationally complex reduction schemes to improve the performance, we adjust the means of the components of the posterior and simply use the component with the largest weight as the state estimator, thus avoiding the unnecessary evaluation of the parameters of the other components. Additionally, through simulations we show that the difference between the distributions of the estimated state in GSF and AMMSE, as well as the true state is not statistically significant. Hence, despite the fact that by changing the gains of individual filters we are deviating from the true posterior, the total state estimation of AMMSE converges to the MMSE state estimation provided by GSF, and the true state.

The rest of this paper is organized as follows: In Section II

the system model is defined and the notations used throughout the paper are introduced. Next, in Section III, we provide the details of GSF. In Section IV, we present our proposed approximate MMSE filter: first, in Section IV-A the gains of individual filters in the proposed method are derived, and then in Section IV-B it is compared with GSF in terms of computational complexity and the convergence of the two filters is analyzed. The simulation results are provided in Section V, comparing AMMSE, GSF, Kalman and Matched filters in terms of estimation accuracy and precision with synthetic data (Section V-A) and experimental data gathered from an indoor localization system (Section V-B), and the convergence⁶ of the state estimations of the filters and the true state is tested with synthetic data. Finally, in Section VI we provide concluding remarks.

II. SYSTEM MODEL

Suppose a discrete-time linear dynamic system, in which the state sequence $\{\mathbf{x}_k, k \in \mathbb{N}\}$, evolves as a first-order Markov process with additive noise. Hence, using state-space representation, the dynamics equation can be written as

$$\mathbf{x}_k = F_k \mathbf{x}_{k-1} + \mathbf{v}_k, \quad (1)$$

where the process noise, $\{\mathbf{v}_k, k \in \mathbb{N}\}$ is an i.i.d. random vector sequence with the pdf $p(\mathbf{v}_k)$ and F_k is a known matrix which describes the linear relationship between the previous and current state. If n_x denotes the dimension of the state vector, the process noise is of dimension n_x and matrix F_k is of size $n_x \times n_x$.

In many applications, the state of the system cannot be observed directly. Hence, it is desirable to estimate the unobservable state from the available measurements. If we denote the measurement sequence by $\{\mathbf{z}_k, k \in \mathbb{N}\}$, the relationship between \mathbf{x}_k and \mathbf{z}_k is described by the measurement equation,

$$\mathbf{z}_k = H_k \mathbf{x}_k + \mathbf{w}_k, \quad (2)$$

where the measurement noise $\{\mathbf{w}_k, k \in \mathbb{N}\}$ is an i.i.d. random vector sequence with the pdf $p(\mathbf{w}_k)$, and it is independent from the process noise. The matrix defining the linear relationship between the current state and measurement vectors, H_k , is a known matrix of size $n_z \times n_x$, where n_z is the dimension of the measurement vector \mathbf{z}_k , and the noise vector \mathbf{w}_k .

Having the above, Bayesian tracking techniques can be used to probabilistically estimate the current state of the system from the available measurements. This is done by recursively estimating the posterior, $p(\mathbf{x}_k | \mathbf{z}_{1:k})$, where $\mathbf{z}_{1:k}$ represents the available measurements up to and including time k . The recursive estimation of the posterior comprises of two steps: prediction and update. In the first step, \mathbf{x}_k is predicted using the previous measurements, i.e. the pdf $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ is estimated using Chapman-Kolmogorov equation on the previous posterior, $p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1})$, and $p(\mathbf{x}_k | \mathbf{x}_{k-1})$. Next, in the update phase, Bayes rule is used to update the prior and evaluate the posterior, $p(\mathbf{x}_k | \mathbf{z}_{1:k})$. The posterior is then used for next iteration estimation. With Gaussian process and

⁵If no reduction scheme is used and the distributions are GMs (not approximated by GMs).

⁶Convergence in distribution is tested using two-sample Kolmogorov-Smirnov non-parametric tests.

measurement noise, the posterior distribution will be Gaussian and its sufficient statistics, i.e. mean and covariance matrix, are optimally tracked by a Kalman filter [1]. Additionally, since the posterior is Gaussian, the filtered mean approximates the state and it is the MMSE estimator [5], [41].

In this work we use GM models for process and measurement noise processes, since they are mathematically tractable and they can be used to approximate any non-Gaussian distribution. Specifically, with GM approximation, the pdf can be approximated as closely as desired [3, Lemma 4.1] and this estimation is asymptotically unbiased [4]. Hence, the process noise, \mathbf{v}_k is estimated by a GM with C_{v_k} components, with $\{\mathbf{u}_k^i, 1 \leq i \leq C_{v_k}\}$ the component means, $\{Q_k^i, 1 \leq i \leq C_{v_k}\}$ the component covariance matrices and $\{w_k^i, 1 \leq i \leq C_{v_k}\}$ the non-negative mixing coefficients. This can be written as:

$$p(\mathbf{v}_k) \approx \sum_{i=1}^{C_{v_k}} w_k^i \mathcal{N}(\mathbf{v}_k; \mathbf{u}_k^i, Q_k^i), \quad (3)$$

where $\sum_{i=1}^{C_{v_k}} w_k^i = 1$, and $\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Sigma)$ represents a Gaussian distribution with argument \mathbf{x} , mean $\boldsymbol{\mu}$, and covariance matrix Σ . The measurement noise distribution can also be approximated by a GM in a similar manner, and written as:

$$p(\mathbf{w}_k) \approx \sum_{j=1}^{C_{w_k}} p_k^j \mathcal{N}(\mathbf{w}_k; \mathbf{b}_k^j, R_k^j), \quad (4)$$

where, C_{w_k} is the number of components of the GM distribution with the non-negative coefficients $\{p_k^j, 1 \leq j \leq C_{w_k}\}$, and $\sum_{j=1}^{C_{w_k}} p_k^j = 1$. The mean and covariance matrix of component $j, 1 \leq j \leq C_{w_k}$ are \mathbf{b}_k^j and R_k^j , respectively.

III. GAUSSIAN SUM FILTERS

With the GM noise distributions, i.e. (3)–(4), the dynamic system defined in (1)–(2), can be described as a *Multiple Model* system, with models $\{M_k^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$, corresponding to the different components or *modes* of the process and measurement noises.⁷ Hence, the posterior can be partitioned as follows:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \sum_{i,j} p(\mathbf{x}_k | \mathbf{z}_{1:k}, M_k^{ij}) p(M_k^{ij} | \mathbf{z}_{1:k}). \quad (5)$$

The mode-conditioned posterior, $p(\mathbf{x}_k | M_k^{ij}, \mathbf{z}_{1:k})$, is a Gaussian distribution with the pdf,

$$p(\mathbf{x}_k | M_k^{ij}, \mathbf{z}_{1:k}) = \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{ij}, \mathbf{P}_{k|k}^{ij}). \quad (6)$$

where its parameters $\hat{\mathbf{x}}_{k|k}^{ij}$ and $\mathbf{P}_{k|k}^{ij}$ can be tracked using the *mode-matched* Kalman filter [1], [5] as follows:

$$\hat{\mathbf{x}}_{k|k-1}^{ij} = F_k \hat{\mathbf{x}}_{k-1|k-1}^{ij} + \mathbf{u}_k^i, \quad (7)$$

$$\mathbf{P}_{k|k-1}^{ij} = Q_k^i + F_k P_{k-1|k-1}^{ij} F_k^\top, \quad (8)$$

⁷For simplicity we assume a system of order one, where all the models have the same history of states. This assumption can be easily relaxed.

$$\boldsymbol{\nu}_k^{ij} = \mathbf{z}_k - (H_k \hat{\mathbf{x}}_{k|k-1}^{ij} + \mathbf{b}_k^j), \quad (9)$$

$$\mathbf{S}_k^{ij} = H_k P_{k|k-1}^{ij} H_k^\top + R_k^j, \quad (10)$$

$$\mathbf{W}_k^{ij} = P_{k|k-1}^{ij} H_k^\top \mathbf{S}_k^{ij-1}, \quad (11)$$

$$\hat{\mathbf{x}}_{k|k}^{ij} = \hat{\mathbf{x}}_{k|k-1}^{ij} + \mathbf{W}_k^{ij} \boldsymbol{\nu}_k^{ij}, \quad (12)$$

$$\mathbf{P}_{k|k}^{ij} = P_{k|k-1}^{ij} - \mathbf{W}_k^{ij} \mathbf{S}_k^{ij} \mathbf{W}_k^{ij\top}. \quad (13)$$

where $(\cdot)^\top$ indicates the transpose of its argument.

Hence, defining

$$\mu_k^{ij} \triangleq p(M_k^{ij} | \mathbf{z}_{1:k}), \quad (14)$$

we can write (5) as a GM distribution, with $C_{v_k} \times C_{w_k}$ components:

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \sum_{i,j} \mu_k^{ij} \mathcal{N}(\mathbf{x}_k; \hat{\mathbf{x}}_{k|k}^{ij}, \mathbf{P}_{k|k}^{ij}). \quad (15)$$

The coefficients μ_k^{ij} can be evaluated as:

$$\begin{aligned} \mu_k^{ij} &= p(M_k^{ij} | \mathbf{z}_k, \mathbf{z}_{1:k-1}) \\ &= \frac{p(\mathbf{z}_k | M_k^{ij}, \mathbf{z}_{1:k-1}) p(M_k^{ij} | \mathbf{z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{z}_{1:k-1})} \\ &= \frac{\Lambda_k^{ij} p(M_k^{ij} | \mathbf{z}_{1:k-1})}{\sum_{lm} \Lambda_k^{lm} p(M_k^{lm} | \mathbf{z}_{1:k-1})}, \end{aligned} \quad (16)$$

where Λ_k^{ij} is the likelihood function and is defined as:

$$\begin{aligned} \Lambda_k^{ij} &\triangleq p(\mathbf{z}_k | M_k^{ij}, \mathbf{z}_{1:k-1}) \\ &= \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_k^{ij}, S_k^{ij}), \end{aligned} \quad (17)$$

and assuming that the current model is independent from the previous model we have⁸

$$p(M_k^{ij} | \mathbf{z}_{1:k-1}) = w_k^i p_k^j. \quad (18)$$

Hence, we can write:

$$\mu_k^{ij} = \frac{w_k^i p_k^j \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_k^{ij}, S_k^{ij})}{\sum_{lm} w_k^l p_k^m \mathcal{N}(\mathbf{z}_k; \hat{\mathbf{z}}_k^{lm}, S_k^{lm})}. \quad (19)$$

A. Reduction Schemes

As mentioned earlier, to avoid an exponentially growing bank size, reduction schemes should be applied to the posterior. In our work we use the less computationally complex schemes: merging all components to their moment-matching Gaussian distribution, and removing the components with smaller weights.⁹

For the first method, the moment-matched Gaussian distribution will have the following mean and covariance matrix:

$$\hat{\mathbf{x}}_{k|k} = \sum_{ij} \mu_k^{ij} \hat{\mathbf{x}}_{k|k}^{ij}, \quad (20)$$

⁸This assumption can be easily relaxed.

⁹Other metrics can be used to determine the active model.

$$P_{k|k} = \sum_{ij} \mu_k^{ij} \left(P_{k|k}^{ij} \right) \quad (21)$$

$$+ \left(\hat{\mathbf{x}}_{k|k}^{ij} - \hat{\mathbf{x}}_{k|k} \right) \left(\hat{\mathbf{x}}_{k|k}^{ij} - \hat{\mathbf{x}}_{k|k} \right)^T \quad (22)$$

$$= \sum_{ij} \mu_k^{ij} \left(P_{k|k}^{ij} + \hat{\mathbf{x}}_{k|k}^{ij} \hat{\mathbf{x}}_{k|k}^{ijT} \right) - \hat{\mathbf{x}}_{k|k} \hat{\mathbf{x}}_{k|k}^T. \quad (23)$$

Alternatively, rather than making a soft decision about the active model, it can be determined by a hard decision. This is the approach used in [6], [7], [17], [20]. A simple scheme to determine the active model is to choose the component with the largest weight. Using this approach, if

$$ij = \arg \max_{lm} \mu_k^{lm}, \quad (24)$$

we have

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k}^{ij}, \quad (25)$$

$$P_{k|k} = P_{k|k}^{ij}. \quad (26)$$

One of the advantages of using this method over replacing the GM with its moment-matched Gaussian distribution, is that by determining the active model before evaluating the parameters of all components, computational resources can be saved. Additionally, evaluating (20)–(23) is more computationally complex than (25)–(26). Besides the computational complexity, by using a soft decision approach, we could be getting drifted from the *matched* estimation. Specifically, at every iteration only one model is active corresponding to the *Matched* filter. By incorporating the outputs of the mismatched filters in (20)–(23), the estimation accuracy and precision are lost. However, if the active model, hence the Matched filter are not chosen correctly, the soft decision method will provide better estimations. For simplicity, we refer to the first method as *merge* and the second as *remove*, e.g. GSF with the first method as the reduction scheme is referred to as *GSF-merge* in this paper.

Additionally, throughout this paper we use the term *mode-matched filter* to refer to the filter estimating the mode-conditioned state, from $p(\mathbf{x}_k | M_k^{ij}, \mathbf{z}_{1:k})$. The term *Matched filter*, is used to denote the mode-matched filter corresponding to the known active model, M_k^* . Having the information about the active model, (5) is simplified and the posterior of the Matched filter will be

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = p(\mathbf{x}_k | \mathbf{z}_{1:k}, M_k^*). \quad (27)$$

Matched filter cannot be implemented without prior knowledge about the active model and is only used for comparison purposes (see Section V-A).

IV. AMMSE ESTIMATOR FOR GM NOISE

In this section we evaluate the gains of individual filters, W_k^{ij} such that the trace of the covariance matrix of the bank of filters, (23) is minimized. Since this covariance matrix is conditional on the measurements sequence, the evaluated gains are functions of innovations.

A. Derivation of Filter Gains

For an arbitrary filter gain W_k^{ij} , the state estimation error covariance matrix for filter ij can be written as follows:¹⁰

$$P_{k|k}^{ij} = P_{k|k-1}^i - W_k^{ij} H_k^T P_{k|k-1}^i - P_{k|k-1}^i H_k^T W_k^{ijT} + W_k^{ij} S_k^{ij} W_k^{ijT} \quad (28)$$

Additionally, using (7) and (12), we can write

$$\hat{\mathbf{x}}_{k|k}^{ij} = F_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{u}_k^i + W_k^{ij} \boldsymbol{\nu}_k^{ij}.$$

Using this in (20), we have:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} &= \sum_{ij} \mu_k^{ij} (F_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbf{u}_k^i) + \mathbb{S}_k \\ &= F_k \hat{\mathbf{x}}_{k-1|k-1} + \mathbb{U}_k + \mathbb{S}_k, \end{aligned} \quad (29)$$

where

$$\mathbb{S}_k \triangleq \sum_{ij} \mu_k^{ij} W_k^{ij} \boldsymbol{\nu}_k^{ij}, \quad (30)$$

$$\mathbb{U}_k \triangleq \sum_{ij} \mu_k^{ij} \mathbf{u}_k^i. \quad (31)$$

Hence,

$$\begin{aligned} \hat{\mathbf{x}}_{k|k}^{ij} \hat{\mathbf{x}}_{k|k}^{ijT} &= F_k \hat{\mathbf{x}}_{k-1|k-1} \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T + F_k \hat{\mathbf{x}}_{k-1|k-1} \mathbf{u}_k^i{}^T \\ &\quad + F_k \hat{\mathbf{x}}_{k-1|k-1} \boldsymbol{\nu}_k^{ijT} W_k^{ijT} + \mathbf{u}_k^i \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T \\ &\quad + \mathbf{u}_k^i \mathbf{u}_k^i{}^T + \mathbf{u}_k^i \boldsymbol{\nu}_k^{ijT} W_k^{ijT} \\ &\quad + W_k^{ij} \boldsymbol{\nu}_k^{ij} \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T + W_k^{ij} \boldsymbol{\nu}_k^{ij} \mathbf{u}_k^i{}^T \\ &\quad + W_k^{ij} \boldsymbol{\nu}_k^{ij} \boldsymbol{\nu}_k^{ijT} W_k^{ijT}, \end{aligned}$$

and

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} \hat{\mathbf{x}}_{k|k}^T &= F_k \hat{\mathbf{x}}_{k-1|k-1} \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T + F_k \hat{\mathbf{x}}_{k-1|k-1} \mathbb{U}_k^T \\ &\quad + F_k \hat{\mathbf{x}}_{k-1|k-1} \mathbb{S}_k^T + \mathbb{U}_k \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T + \mathbb{U}_k \mathbb{U}_k^T \\ &\quad + \mathbb{U}_k \mathbb{S}_k^T + \mathbb{S}_k \hat{\mathbf{x}}_{k-1|k-1}^T F_k^T + \mathbb{S}_k \mathbb{U}_k^T + \mathbb{S}_k \mathbb{S}_k^T. \end{aligned}$$

To find the optimal W_k^{ij} we need to have:

$$\frac{\partial \text{tr}(P_{k|k})}{\partial W_k^{ij}} = 0. \quad (32)$$

Thus, using the above and (28) in (23), we can write:¹¹

$$\begin{aligned} &-P_{k|k-1}^i H_k^T + W_k^{ij} S_k^{ij} \\ &+ \mathbf{u}_k^i \boldsymbol{\nu}_k^{ijT} + W_k^{ij} \boldsymbol{\nu}_k^{ij} \boldsymbol{\nu}_k^{ijT} \\ &- \mathbb{U}_k \boldsymbol{\nu}_k^{ijT} - \mathbb{S}_k \boldsymbol{\nu}_k^{ijT} = 0 \end{aligned} \quad (33)$$

and we will have:

$$\begin{aligned} W_k^{ij} &= \left(P_{k|k-1}^i H_k^T + \mathbb{U}_k \boldsymbol{\nu}_k^{ijT} - \mathbf{u}_k^i \boldsymbol{\nu}_k^{ijT} + \mathbb{S}_k \boldsymbol{\nu}_k^{ijT} \right) \\ &\quad \times \left(S_k^{ij} + \boldsymbol{\nu}_k^{ij} \boldsymbol{\nu}_k^{ijT} \right)^{-1} \\ &= A_k^{ij} + \mathbb{S}_k B_k^{ij}, \end{aligned} \quad (34)$$

¹⁰For Kalman gain this equation is simplified as in (11).

¹¹We assume $\mu_k^{ij} \neq 0$, since for $\mu_k^{ij} = 0$ there is no need to evaluate the parameters of filter ij .

where

$$A_k^{ij} \triangleq \left(P_{k|k-1}^i H_k^T + U_k \nu_k^{ijT} - \mathbf{u}_k^i \nu_k^{ijT} \right) \times \left(S_k^{ij} + \nu_k^{ij} \nu_k^{ijT} \right)^{-1}, \quad (35)$$

$$B_k^{ij} \triangleq \nu_k^{ijT} \left(S_k^{ij} + \nu_k^{ij} \nu_k^{ijT} \right)^{-1}. \quad (36)$$

Using (30), (34), \mathbb{S}_k can be evaluated as follows:

$$\sum_{ij} \mu_k^{ij} W_k^{ij} \nu_k^{ij} = \sum_{ij} \mu_k^{ij} A_k^{ij} \nu_k^{ij} + \mathbb{S}_k \sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij}. \quad (37)$$

Since the term $\sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij}$ is a scalar, we can write

$$\mathbb{S}_k \left(1 - \sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij} \right) = \sum_{ij} \mu_k^{ij} A_k^{ij} \nu_k^{ij}, \quad (38)$$

thus,

$$\mathbb{S}_k = \sum_{ij} \mu_k^{ij} A_k^{ij} \nu_k^{ij} \left(1 - \sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij} \right)^{-1}. \quad (39)$$

This can be used in (34) to determine the optimal gain for filter ij .

Having the above, the estimated state, $\hat{\mathbf{x}}_{k|k}$ minimizing the trace of the covariance matrix of the filter can be evaluated from (29). However, changing the parameters of the components in the posterior, leads to larger MSE when compared with GSF, the MMSE filter.¹² Specifically, the MSE of AMMSE filter can be evaluated as follows:

$$\text{MSE}^{\text{AMMSE}} \triangleq \mathbb{E}_{\mathbf{x}_k} \left\{ (\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}) (\mathbf{x}_k - \hat{\mathbf{x}}_{k|k})^T | \mathbf{z}_{1:k} \right\} \quad (40)$$

$$= P_{k|k} + (\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k}) (\hat{\mathbf{x}}_{k|k} - \hat{\mathbf{x}}_{k|k})^T \quad (41)$$

where $\mathbb{E}_{\mathbf{x}}\{g(\mathbf{x})\}$ is the expected value of the function $g(\mathbf{x})$ with respect to the random variable \mathbf{x} with pdf $p(\mathbf{x})$, and $\hat{\mathbf{x}}_{k|k}$, $P_{k|k}$ are the parameters of the MMSE filter evaluated as in (20)–(23).

B. Comparison with GSF

In this section we compare GSF and the proposed AMMSE filter in terms of computational complexity for one iteration (Section IV-B1) and discuss the convergence of AMMSE estimator to GSF (Section IV-B2).

1) *Computational Complexity*: In the first stage of Bayesian tracking, i.e. prediction, GSF and AMMSE have the same steps in (7)–(10), (17), (19), hence the same computational complexity. The update stage can be divided into three main operations: evaluating the gains, the component means, and covariance matrices. However, depending on the reduction scheme used, the number of times that these parameters have to be evaluated changes. Specifically, when the components are merged, the parameters of all $C_{v_k} C_{w_k}$ filters in the banks have to be evaluated, whereas when the components with smaller

weights are removed, only the parameters of the filter with the maximum weight need to be computed. Moreover, merging the means and covariance matrices of the components in (20)–(21) is more computationally complex than simply removing the components with smaller weights in (25)–(26). In Table I, we provide the computational complexity of AMMSE-merge/-remove and GSF-merge/-remove for the update stage and reduction scheme. The matrix inversions are not considered in this table for two reasons: 1) The three filters with comparable performances, i.e. AMMSE-merge/-remove and GSF-merge all have the same computational complexity in this respect, specifically, $C_{v_k} C_{w_k}$ inversions of matrices of size $n_z \times n_z$, and although GSF-remove requires only one such inversion, its performance is not comparable to the others; 2) Even by not considering the matrix inversions, GSF-remove has the lowest computational complexity and the table serves its purpose of comparison. The detailed number of operations required for evaluating all parameters are given in Tables VII–IX in Appendix A.

When merging is used as a reduction scheme, both GSF and AMMSE have the same computational complexity for evaluating the gains, component means and covariance matrices, as well as merging the moments of the individual filters. For GSF-remove, each parameter needs to be evaluated for the *correct* component only. However, this is not true for AMMSE-remove, as the gains are dependent. Thus, to compute \mathbb{S}_k , the parameters A_k^{ij} have to be evaluated for all filters. Having \mathbb{S}_k , the parameters of the component with the maximum weight can be computed with the same computational complexity as GSF-remove. Additionally, the number of matrix inversions for GSF-remove is $C_{v_k} C_{w_k}^{-1}$ times the number of inversions for all the other three filters. Hence, GSF-remove has the lowest computational complexity. However, as shown in Section V this is at the cost of losing estimation precision and accuracy. The other three filters, i.e. AMMSE-merge/-remove and GSF-merge show similar performance with AMMSE-remove requiring the least number of operations.

2) *Convergence to MMSE Estimator*: At iteration k , the AMMSE state estimation, $\hat{\mathbf{x}}_{k|k}$, converges in distribution to the MMSE state estimation provided by GSF, $\hat{\mathbf{x}}_{k|k}$. Using (40), the convergence of the state estimations yields the convergence of the MSEs. Hence, the MSE of the AMMSE filter converges to the MSE of GSF-merge. Since it is not easily feasible to provide an analytical proof for all cases due to the matrix inversions in the evaluation of gains, we prove the convergence for two limit cases in Appendix B, and use Kolmogorov-Smirnov test for the other cases in Section V.

In the analytical proof in Appendix B, the convergence of state estimations is proved by showing the convergence of the parameters of the GM posterior, namely the coefficients, μ_k^{ij} , the means, $\hat{\mathbf{x}}_{k|k}^{ij}$, and the covariance matrices, $P_{k|k}^{ij}$, of the components. Since these parameters are dependent on the innovations of individual filters, ν_k^{ij} , two limit cases are considered: when the distance between innovations approaches zero and when it goes to infinity. The first case applies to a posterior with highly overlapping components, where the likelihoods of all models are close to one. Contrarily, when the distance between innovations approaches infinity, the

¹²GSF is the MMSE estimator when no reduction scheme is used and the noise distributions are GMs, rather than being *approximated* by GMs. However, as shown in [42] even with mixture reduction, GSF-merge converges to the MMSE filter.

TABLE I
COMPUTATIONAL COMPLEXITY OF GSF-MERGE/-REMOVE, AND AMMSE-MERGE/-REMOVE

Filter	Parameter	Computational complexity
GSF-merge	$\{W_k^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x^2 n_z)$
	$\{P_{k k}^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x^2 n_z)$
	$\{\hat{x}_{k k}^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x n_z)$
	$\hat{x}_{k k}, P_{k k}$	$O(C_{v_k} C_{w_k} n_x^2)$
AMMSE-merge	$\{W_k^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x^2 n_z)$
	$\{P_{k k}^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x^2 n_z)$
	$\{\hat{x}_{k k}^{ij}; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}\}$	$O(C_{v_k} C_{w_k} n_x n_z)$
	$\hat{x}_{k k}, P_{k k}$	$O(C_{v_k} C_{w_k} n_x^2)$
GSF-remove	$\{W_k^{ij}; ij = \arg \max_{lm} \mu_k^{lm}\}$	$O(n_x^2 n_z)$
	$\{P_{k k}^{ij}; ij = \arg \max_{lm} \mu_k^{lm}\}$	$O(n_x^2 n_z)$
	$\{\hat{x}_{k k}^{ij}; ij = \arg \max_{lm} \mu_k^{lm}\}$	$O(n_x n_z)$
AMMSE-remove	\mathbb{S}_k	$O(C_{v_k} C_{w_k} n_x^2 n_z)$
	$\{P_{k k}^{ij}; ij = \arg \max_{lm} \mu_k^{lm}\}$	$O(n_x^2 n_z)$
	$\{\hat{x}_{k k}^{ij}; ij = \arg \max_{lm} \mu_k^{lm}\}$	$O(n_x n_z)$

likelihood of the active model is close to one and all the other components have negligible likelihoods. In other words, the active model can be well determined by using the coefficients, μ_k^{ij} , of the GM posterior.

For general GM noise models, the analytical proof is not straight forward, due to the matrix inversions in the gains, $(S_k^{ij} + \nu_k^{ij} \nu_k^{ij\top})^{-1}$ and S_k^{ij-1} . Hence, in Section V, we provide Kolmogorov-Smirnov (KS) statistic for the distributions, $p(\hat{x}_{k|k})$ and $p(\hat{x}_{k|k})$ under different types of GM noise parameters. Specifically, we use GM noise models with different separations between components, and by running simulations on synthetically generated data with these noise models, we generate samples from the distributions $p(\hat{x}_{k|k})$ and $p(\hat{x}_{k|k})$. Using the KS test on these data, the hypothesis that the two data samples belong to the same distribution, is accepted at %95 confidence interval.

V. SIMULATION RESULTS

We consider two simulation scenarios: In the first scenario, we use synthetically generated process and measurement noise processes, whereas for the second simulation scenario we gather experimental data from an indoor localization system with ultra-wideband (UWB) sensors. To be consistent, we use the same process and measurement equations for both scenarios and they only differ in the noise distributions.

For both scenarios, we consider an indoor localization problem, in a 2D setting, and track the position in each direction independently assuming noise distributions with time-invariant statistics. The state vector contains the position and the velocity, but only noisy information about the position is observable and measured. Hence, in each direction we have $n_x = 2$, $n_z = 1$, and

$$F_k = \begin{bmatrix} 1 & \Delta t_k \\ 0 & 1 \end{bmatrix}, \quad (42)$$

$$H_k = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad (43)$$

where Δt_k is the time interval between the measurements z_{k-1} and z_k . In our localization system, the time intervals between measurements are multiples of 0.1080 s. Hence, in our synthetic setting, we use $\Delta t_k = 0.1080$ s for all iterations, k .

We use a random walk velocity motion model,

$$\mathbf{v}_k = v_k \times \begin{bmatrix} \Delta t_k \\ 1 \end{bmatrix}, \quad (44)$$

where v_k is a univariate GM random variable with C_{v_k} components, with $\{u_k^i; 1 \leq i \leq C_{v_k}\}$ the component means, $\{\sigma_k^i; 1 \leq i \leq C_{v_k}\}$ the component variances. Hence, in (3) we have:

$$\forall i; 1 \leq i \leq C_{v_k}, \mathbf{u}_k^i = u_k^i \times \begin{bmatrix} \Delta t_k \\ 1 \end{bmatrix}, \quad (45)$$

$$Q_k^i = \sigma_k^i{}^2 \times \begin{bmatrix} \Delta t_k^2 & \Delta t_k \\ \Delta t_k & 1 \end{bmatrix}. \quad (46)$$

For the experimental setup, the noise distributions, $p(v_k)$ and $p(w_k)$ are estimated using the data gathered from the UWB sensors. For the synthetically generated data, we assume the same distribution for process and measurement noise, i.e. $p(v_k) = p(w_k)$.

The following filtering schemes are used and compared:

- 1) Kalman filter (KF)
- 2) GSF-merge/remove
- 3) AMMSE-merge/remove
- 4) Matched filter (for synthetically generated data)

The performance metrics that we used for analyzing the estimation accuracy and precision are root-mean-square error (RMSE) and circular error probable (CEP), respectively. Denoting the location estimation error at iteration k by ϵ_k , the RMSE and CEP are defined as:

$$\text{RMSE} \triangleq \left(\frac{1}{N} \sum_{k=1}^N \epsilon_k^2 \right)^{1/2}, \quad (47)$$

TABLE II
THE PARAMETERS OF THE GM MODELS USED FOR GENERATING
SYNTHETIC DATA

Model 1	$\mathbf{w} = [0.2, 0.2, 0.2, 0.2, 0.2]$ $\mathbf{m} = c[-50, -30, 0, 30, 50]$
Model 2	$\mathbf{w} = [0.1, 0.1, 0.6, 0.1, 0.1]$ $\mathbf{m} = c[-50, -30, 0, 30, 50]$
Model 3	$\mathbf{w} = [0.5, 0.1, 0.1, 0.1, 0.2]$ $\mathbf{m} = c[-50, 10, 30, 50, 80]$

$$\text{CEP} \triangleq F_{|\epsilon|}^{-1}(0.5), \quad (48)$$

where N is the total number of iterations and $F_{|\epsilon|}^{-1}$ represents the inverse cumulative distribution function of error evaluated over the whole experiment.

A. Synthetic data

For the synthetically generated data, the process and measurement noises are assumed to be i.i.d. samples from one of the following GM models:

Model 1: Symmetric distribution with the components all having the same coefficients.

Model 2: Symmetric distribution with components possibly having different weights.

Model 3: Asymmetric distribution.

The coefficients of the components are chosen such that the noises are zero mean. In our simulations, we assumed GM distributions with 5 components, each having a variance of 1, i.e. we have

$$C_{v_k} = C_{w_k} = 5, \quad (49)$$

$$\forall i; 1 \leq i \leq 5, \sigma_k^{i2} = 1, \quad (50)$$

$$\forall j; 1 \leq j \leq 5, R_k^j = 1. \quad (51)$$

The coefficients, \mathbf{w} , and the means, \mathbf{m} , of the components are given in Table II, where

$$\mathbf{w} = [w_k^1, \dots, w_k^5] = [p_k^1, \dots, p_k^5], \quad (52)$$

$$\mathbf{m} = [b_k^1, \dots, b_k^5] = [u_k^1, \dots, u_k^5]. \quad (53)$$

The parameter c is multiplied by the means to change the distance between components.

In addition to the above models, to provide more general simulation results we also used the noise models from [19, Section V-B-C] corresponding to equal means and unequal covariance matrices. The first scenario corresponds to a maneuvering target tracking problem with GM process noise and Gaussian measurement noise. The second example corresponds to a system with Gaussian process noise and glint measurement noise. The parameters of the used models are provided in Table III. In this table, we use $[a^i]_1^n$ to denote $[a^1, \dots, a^n]$.

Using the GM noise models in Table II–III, we generated the measurements z_k and estimated the state with Kalman filter, GSF-merge/-remove, and AMMSE-merge/-remove, as well as the Matched filter. To do this, we label the generated data by the active noise models in effect, and use this information to choose the correct filter in the bank to achieve the Matched filter. To further investigate the effect of multi-modality on

TABLE III
THE PARAMETERS OF THE TWO SCENARIOS FROM [19]

Maneuvering target	$C_{v_k} = 2$ $[w_k^i]_1^2 = [0.8, 0.2]$ $[u_k^i]_1^2 = [0, 0]$ $[\sigma_k^{i2}]_1^2 = [0.01, 1]$ $C_{w_k} = 1, b_k = 0, R_k = 0.1$
Glint measurement noise	$C_{v_k} = 1, u_k = 0, \sigma_k^2 = 1$ $C_{w_k} = 2$ $[p_k^j]_1^2 = [0.1, 0.9]$ $[b_k^j]_1^2 = [0, 0]$ $R_k^j = [0.01, 1]$

the performance of these filters, we vary the parameter c for the noise models in Table II and evaluate the state estimate, RMSE, and CEP for all filters. For each value of c , 1000 Monte-Carlo runs are used to estimate the RMSE, and CEP at %95 confidence interval. We also approximate the KL divergence between the GM noise distribution and their corresponding moment-matched Gaussian density for each value of c . The state estimations evaluated in these simulations are then used as samples from the pdfs $p(\hat{\mathbf{x}}_{k|k})$, and $p(\hat{\mathbf{x}}_{k|k})$ to approximate the Kolmogorov-Smirnov (KS) statistic and test the hypothesis that the two sets of samples have the same distribution. The results for GM noise distribution with Model 1, 2, and 3 in Table II are provided in Fig. 1, Fig. 2, and Fig. 3, respectively. The results for the maneuvering target tracking scenario and the system with glint measurement noise with the parameters defined in Table III are provided in Table IV. These results show the RMSE and CEP of the filters for different values of KL divergence between the used noise model and its moment-matched Gaussian distribution. As can be expected, as the divergence between the noise distribution and its corresponding fitted Gaussian increases, the performance of Kalman filter drops both in terms of RMSE and CEP. This is due to the fact that Kalman filter is best suited for systems with Gaussian distributions, and it fails to provide good estimations for multi-modal noise models. Additionally, since the component variances of noise distributions are the same and by varying c only the separation between the components changes, the performance of the Matched filter remains the same for all values of divergence. However, for GSF-merge/-remove, and AMMSE-merge/-remove, the performance changes when the KL divergence increases.

With all noise models, the performance of all filters are close to the Matched filter for small KL divergences (as shown in Section IV-B2). However, as the divergence increases, both RMSE and CEP increase for all filters, until they reach their maximum. This is due to the fact that with an increase in the separation between the noise components, hence, the KL divergence with the corresponding moment-matched Gaussian distribution, the overlap between the components is decreasing. However, this decrease is not enough for the filters to correctly find the active model by the weights of components. In other words, the posterior is multi-modal, hence it cannot be well approximated by a single Gaussian. However, the overlap between the components of the GM posterior leads to

drifting from the Matched filter estimation. Further increase in the divergence results in improved performance for AMMSE-merge/-remove and GSF-merge until they converge to the Matched filter (as shown in Section IV-B2). However, this is not the case for GSF-remove. Specifically, as the KL divergence increases, the RMSE and CEP of GSF-remove decrease until they reach their minimum. But further increase in the KL divergence results in increased RMSE and CEP for this filter. This is because with GSF-remove the correct choice of the active component is of particular importance. Although, with the increase in KL divergence the component with the maximum weight represents the active model most of the times, when it fails and a wrong component is chosen, there will be a larger error due to the increased distance between the component corresponding to the active model and the other components.

For all models, the performances of GSF-merge and AMMSE-merge are very close, especially, for KL divergence values greater than 1. To further investigate the relationship between these two filters, we used the state estimations, $\hat{\mathbf{x}}_{k|k}$, and $\hat{\mathbf{x}}_{k|k}$ to test the hypothesis that they come from the same distribution. Using KS test, this hypothesis is accepted at %95 confidence interval for all noise models and KL divergences. Additionally, since GSF-merge is not the MMSE filter due to the reduction of GM posteriors, we also carried out the two-sample KS hypothesis tests on the state estimations from AMMSE-merge and the true state. The hypotheses that the samples have the same distribution are accepted at %95 confidence interval for all noise models for KL divergences greater than 1. Moreover, to test the variances of the two filters we used Ansari-Bradley¹³ test with the null hypothesis that the variances of these two sample sets are equal. The null hypotheses for Ansari-Bradley tests were also accepted for all noise models and KL divergences greater than 0.04 at %95 confidence interval.

The RMSE and CEP of AMMSE-remove are always smaller or equal to the RMSE and CEP of GSF-remove. This is due to the fact that with AMMSE, the means of components are evaluated such that the trace of total state estimation error covariance matrix including the spread of means in (22) is minimized. Contrarily, in GSF the component means, $\hat{\mathbf{x}}_{k|k}^{ij}$ are minimizing the trace of the covariance matrix of each individual filter $P_{k|k}^{ij}$. Hence, the component means are closer in AMMSE filter. Consequently, when removing the components with smaller weights as a reduction scheme, the performance (both accuracy and precision) is better.

Based on the results, it is evident that when the noise distributions are far from Gaussian (the KL divergence between the noise distribution and the corresponding moment-matched Gaussian density is large), GSF-merge, and AMMSE-merge/-remove perform similarly regardless of the shape of the noise model in terms of symmetry. This is particularly important, since among these filters which achieve comparable estimation accuracy and precision, AMMSE-remove requires the least number of operations by avoiding the evaluation of $P_{k|k}^{ij}$ (as

¹³Since the posteriors are not Gaussian the chi-square tests for normalized estimation error squares (NEES) cannot be applied.

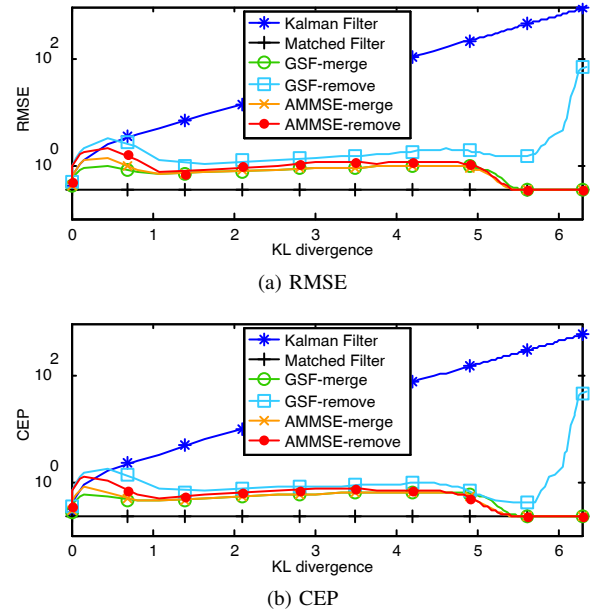


Fig. 1. RMSE and CEP for the synthetic data generated using Model 1 vs. KL divergence between the noise distribution and the moment-matched Gaussian pdf.

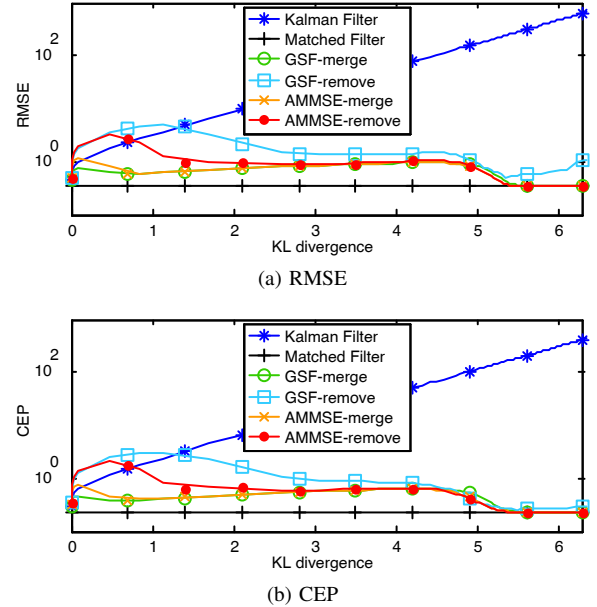


Fig. 2. RMSE and CEP for the synthetic data generated using Model 2 vs. KL divergence between the noise distribution and the moment-matched Gaussian pdf.

shown in Section IV-B1).

B. Experimental Data: Indoor Positioning System

In this section we provide the simulation results of the experimental data gathered from an indoor positioning system with off-the-shelf *Ubisense* UWB location sensors. In our setting, we have four UWB receivers, four stationary objects and a mobile target. A UWB tag is attached to each stationary and mobile object and sends UWB pulses to the four receivers. The system estimates the location of these tags by computing the time and angle difference of arrival of their UWB pulses.

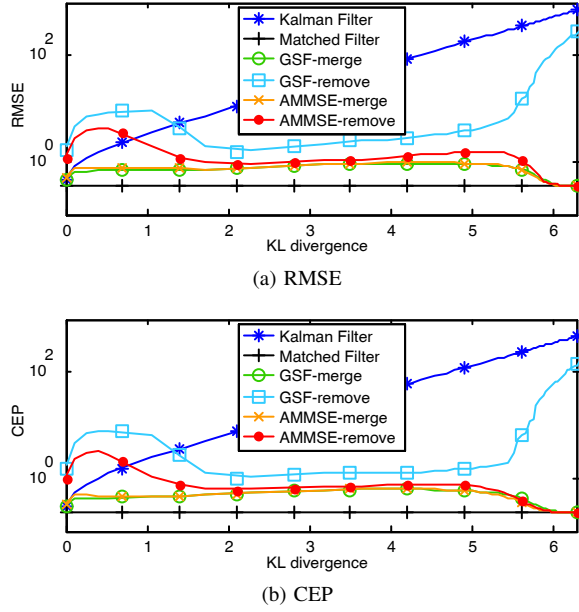


Fig. 3. RMSE and CEP for the synthetic data generated using Model 3 vs. KL divergence between the noise distribution and the moment-matched Gaussian pdf.

TABLE IV
SIMULATION RESULTS FOR MANEUVERING TARGET TRACKING AND GLINT MEASUREMENT NOISE

	Maneuvering target KL div = 0.0155		Glint noise KL div = 0.0042	
	RMSE	CEP	RMSE	CEP
Kalman	0.1204	0.0802	0.3356	0.2191
Matched filter	0.1184	0.0780	0.2274	0.1208
GSF-merge	0.1204	0.0802	0.3224	0.2007
GSF-remove	0.1866	0.1325	0.3354	0.2195
AMMSE-merge	0.1204	0.0802	0.3312	0.2158
AMMSE-remove	0.1860	0.1306	0.3343	0.2187

The location information of the stationary objects is used for approximating the measurement noise distribution by GMs. The histograms of measurement noise in x and y directions are given in Fig. 4. To find the process noise pdfs, we used the ground truth about the location of the mobile target. The experiment is carried out 20 times with the mobile target moving on a predefined trajectory. The histograms of the process noise are depicted in Fig. 4 for x and y directions.

With GM approximations for process and measurement noise, the position is estimated by post-processing the sensor measurements. The approximated parameters of the GM noise distributions in both directions, and also the KL divergence with their corresponding moment-matched Gaussian distributions are given in Table V. In this table, we use $[a^i]_1^n$ to denote $[a^1, \dots, a^n]$.

Fig. 5 shows the location estimations in x and y directions for one of our experiments. To better compare the performance of different filtering schemes, in Table VI we provide the RMSE and CEP in each direction. These values are the average of RMSE and CEP over all our experiments. Since the noise distributions are very close to their moment-matched Gaussian distribution (Table V), the RMSE and CEP of Kalman estimations are very close to the RMSE and CEP of GSF-merge

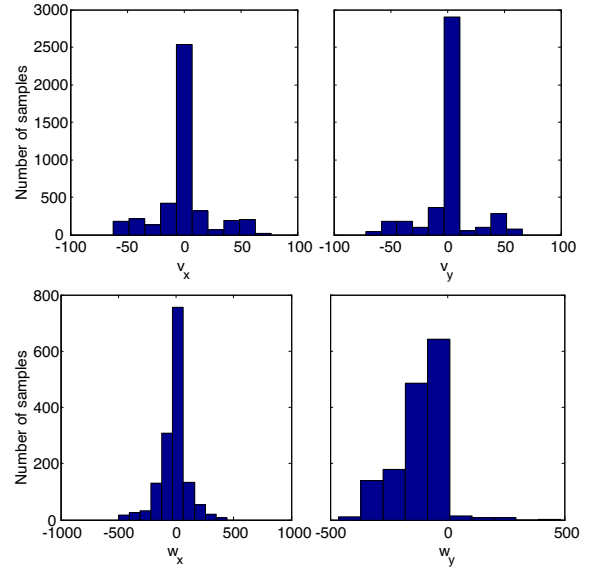


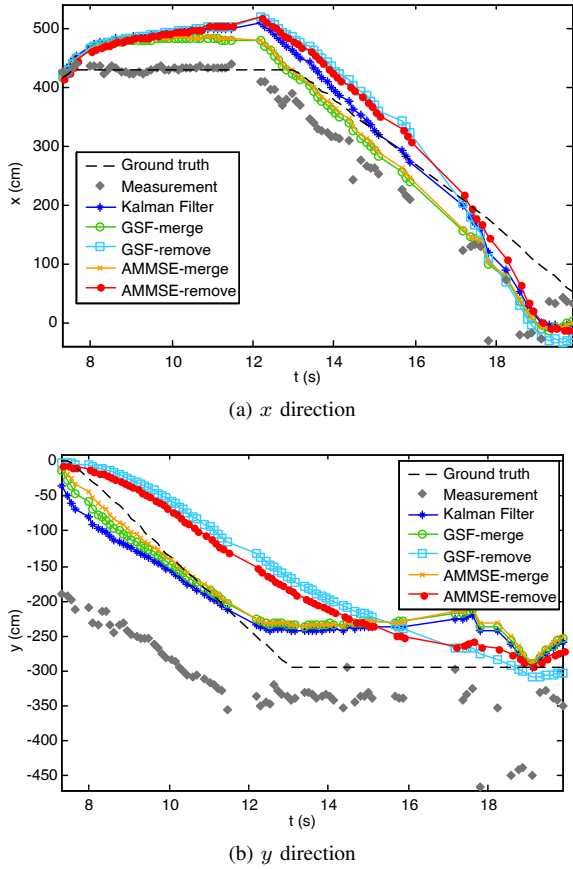
Fig. 4. Histogram of process and measurement noise in x and y directions

TABLE V
THE PARAMETERS OF THE GM NOISE DISTRIBUTIONS

Noise	KL divergence	Parameters
x	0.4253	$C_{v_k} = 3$ $[w_k^i]_1^3 = [0.13, 0.77, 0.099]$ $[u_k^i]_1^3 = [-41.44, 0.51, 49.79]$ $[\sigma_k^{i2}]_1^3 = [148.24, 48.38, 83.75]$
		$C_{w_k} = 3$ $[p_k^j]_1^3 = [0.07, 0.85, 0.08]$ $[b_k^j]_1^3 = [-300.01, -17.06, 207.37]$ $[R_k^j]_1^3 = [8163.20, 3611.99, 5677.21]$
y	1.1971	$C_{v_k} = 9$ $[w_k^i]_1^9 = [0.01, 0.06, 0.03, 0.03, 0.72, 0.04, 0.02, 0.06, 0.03]$ $[u_k^i]_1^9 = [-63.38, -48.73, -35.65, -17.40, -0.32, 9.52, 30.09, 44.24, 54.35]$ $[\sigma_k^{i2}]_1^9 = [24.34, 21.53, 18.18, 23.62, 3.13, 12.16, 18.81, 12.96, 15.44]$
		$C_{w_k} = 2$ $[p_k^j]_1^2 = [0.98, 0.02]$ $[b_k^j]_1^2 = [-125.93, 147.25]$ $[R_k^j]_1^2 = [8500.19, 10809.10]$

and AMMSE-merge. However, by minimizing the trace of the total state estimation error covariance matrix and decreasing the spread of means, AMMSE provides the best results in both directions and the performance of AMMSE-remove is better than GSF-remove.

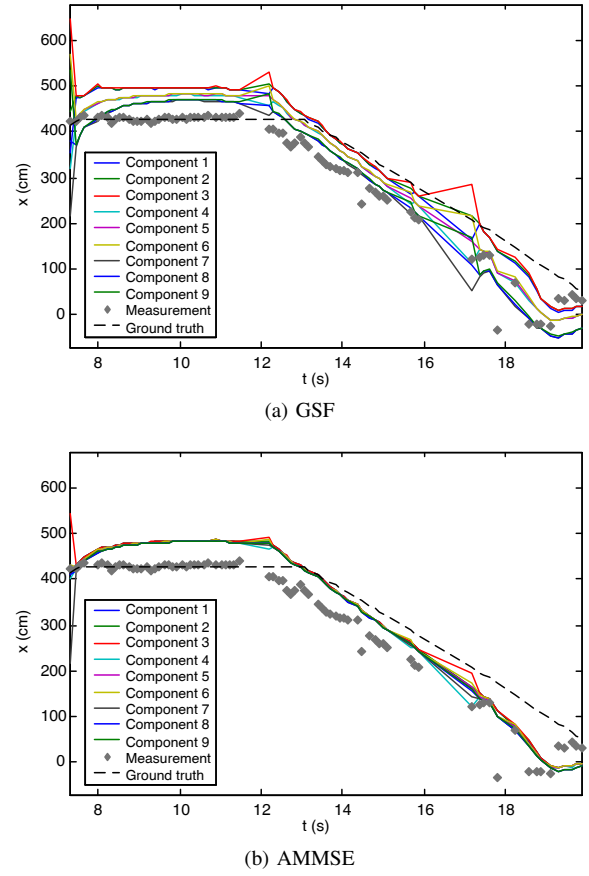
To better depict the difference in spread of means of GSF and AMMSE, their component means are depicted in Fig. 6. This figure shows the component means for one of our experiments in x direction. As shown in this figure, the spread of means in GSF is larger than that of AMMSE. This is especially evident at times $t = 12.19$ s and $t = 17.17$ s.

Fig. 5. Location estimation in x and y directionsTABLE VI
SIMULATION RESULTS FOR EXPERIMENTAL DATA

	x		y	
	RMSE	CEP	RMSE	CEP
Kalman	88.1850	66.3306	73.0512	60.4984
GSF-merge	85.7485	55.2261	74.3982	61.7358
GSF-remove	110.4006	91.7458	75.1093	48.6651
AMMSE-merge	80.7388	52.2901	71.3662	56.7287
AMMSE-remove	101.8026	83.2669	72.0491	44.7549

VI. CONCLUSION

In this paper, we propose an approximate MMSE (AMMSE) state estimator, for linear dynamic systems with Gaussian Mixture (GM) noise. For this purpose, we used a bank of Kalman filters with adjusted gains to track the models corresponding to the different components of the GM noises. This is done by minimizing the trace of the total state estimation error covariance matrix, including the individual filters covariance matrices and the spread of their means. Hence, comparing with Gaussian Sum Filter (GSF) which minimizes the trace of the individual filters covariance matrices by using parallel Kalman filters, our proposed method has a smaller spread of means, and is more robust to removing components. Specifically, we have shown through simulations that unlike GSF, the performance of the proposed AMMSE filter does not change when instead of merging all components, we reduce the number of components by taking the component with the maximum weight. This is specifically important for applications which require lower computational complexities.

Fig. 6. Component centers for GSF and AMMSE in x direction

We have also shown that the distributions of state estimations with GSF and AMMSE filter converge at two limit cases: when the distance between the GM components approach zero and infinity. For the other cases, this is tested with Kolmogorov-Smirnov test on the state estimations of the two filters.

APPENDIX A

NUMBER OF REQUIRED OPERATIONS FOR THE FILTERS

In this section we provide the number of required operations for GSF (Table VII), Approximate MMSE (Table VIII) and merging the means and covariance matrices of individual filters (Table IX). In the first column of these tables, we indicate the required operations for evaluating each parameter. The last column represents the number of times that a certain parameter has to be evaluated. For instance, evaluating the gains has to be done for every filter whereas the value \mathbb{S}_k is only evaluated once for every iteration. However, with remove as the reduction scheme, the number of filters becomes one, and each row in Tables VII–VIII is evaluated only once.

To further compare the two reduction algorithms in terms of computational complexity, we provide the number of required operations for merging the components in Table IX. For simplicity, we use the GSF parameters in this table, however, it applies to both filters.

In these tables, we assume that multiplying $A_{a \times b}$ by $B_{b \times c}$, where the subscripts represent the sizes of the matrices, requires $ac(b - 1)$ additions and abc multiplications. Additionally, we assume that the transposition of matrices requires

no operations and the inversion of a scalar is similar to a multiplication.

APPENDIX B PROOF OF CONVERGENCE

In this section we show that the estimated GM posteriors of AMMSE and GSF converge for two limit cases for the distance between innovations of individual filters. The convergence is proved by showing the convergence of the parameters of the GM posteriors.

(a) When the distances between innovations approach zero, i.e.

$$\forall i \neq l, j \neq m; 1 \leq i, l \leq C_{v_k}, 1 \leq j, m \leq C_{w_k},$$

$$\left\| \nu_k^{ij} - \nu_k^{lm} \right\| \rightarrow 0. \quad (54)$$

In this case, all the innovations are converging to the same value,

$$\forall i, j; 1 \leq i \leq C_{v_k}, 1 \leq j \leq C_{w_k}, \quad \nu_k^{ij} \rightarrow \nu_k^*. \quad (55)$$

Hence, we have

$$\mathbb{S}_k \rightarrow \sum_{ij} \mu_k^{ij} W_k^{ij} \nu_k^*. \quad (56)$$

Now, using (33), we can write:

$$-\sum_{ij} \mu_k^{ij} P_{k|k-1}^i H_k^T + \sum_{ij} \mu_k^{ij} W_k^{ij} S_k^{ij} \rightarrow 0, \quad (57)$$

which holds for GSF gains, W_k^{ij} in (11). But since there is only one unique solution for the set of equalities given in (33) for all i, j , we have

$$W_k^{ij} \rightarrow W_k^{ij}, \quad (58)$$

and consequently,

$$\hat{\mathbf{x}}_{k|k}^{ij} \rightarrow \hat{\mathbf{x}}_{k|k}^{ij}, \quad (59)$$

$$P_{k|k}^{ij} \rightarrow P_{k|k}^{ij}. \quad (60)$$

(b) When the distances between innovations approach infinity¹⁴, i.e.

$$\forall i \neq l, j \neq m; 1 \leq i, l \leq C_{v_k}, 1 \leq j, m \leq C_{w_k},$$

$$\left\| \nu_k^{ij} - \nu_k^{lm} \right\| \rightarrow \infty. \quad (61)$$

In this case, GSF and AMMSE filter converge to the Matched filter. Specifically, if we denote the active model by M_k^* , we will have

$$\nu_k^* \rightarrow 0, \quad (62)$$

hence,

$$\Lambda_k^* \rightarrow 1. \quad (63)$$

On the other hand, for all the other models $\{M_k^{lm}; 1 \leq l \leq C_{v_k}, 1 \leq m \leq C_{w_k}, M_k^{lm} \neq M_k^*\}$, the

¹⁴We have assumed that the filters (GSF/AMMSE) are stable and have bounded error. Specifically, by choosing the wrong components, the state estimation error can increase for all individual filters in the bank leading to all the innovations approaching infinity. This is not the case considered here.

innovations ν_k^{lm} increase and approach infinity, hence we will have

$$\Lambda_k^{lm} \rightarrow 0. \quad (64)$$

Thus, using (16), we will have

$$\mu_k^* \rightarrow 1, \quad (65)$$

$$\mu_k^{lm} \rightarrow 0. \quad (66)$$

Hence, after merging¹⁵

$$\hat{\mathbf{x}}_{k|k} \rightarrow \hat{\mathbf{x}}_{k|k}^*, \hat{\mathbf{x}}_{k|k} \rightarrow \hat{\mathbf{x}}_{k|k}^*, \quad (67)$$

$$P_{k|k} \rightarrow P_{k|k}^*, P_{k|k} \rightarrow P_{k|k}^*. \quad (68)$$

Moreover, using (35)–(36), we will have

$$A_k^* \rightarrow W_k^{ij}, \quad (69)$$

$$B_k^* \rightarrow 0, \quad (70)$$

whereas for the non-matching models, $\{M_k^{lm}; 1 \leq l \leq C_{v_k}, 1 \leq m \leq C_{w_k}, M_k^{lm} \neq M_k^*\}$,

$$A_k^{lm} \rightarrow 0, \quad (71)$$

$$B_k^{lm} \rightarrow 0. \quad (72)$$

Thus,

$$\hat{\mathbf{x}}_{k|k}^* \rightarrow \hat{\mathbf{x}}_{k|k}^*, \quad (73)$$

$$P_{k|k}^* \rightarrow P_{k|k}^*. \quad (74)$$

ACKNOWLEDGMENT

This work was partly supported by the Natural Sciences and Engineering Research Council (NSERC) and industrial and government partners, through the Healthcare Support through Information Technology Enhancements (hSITE) Strategic Research Network.

REFERENCES

- [1] Y.-C. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 333–339, 1964.
- [2] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [3] B. D. O. Anderson and J. B. Moore, *Optimal filtering*. Englewood Cliffs: Prentice-Hall, 1979.
- [4] E. Parzen, "On estimation of a probability density function and mode," *The Annals of Mathematical Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.
- [5] Y. Bar-Shalom, X. Li, and T. Kirubarajan, *Estimation with applications to tracking and navigation: theory algorithms and software*. Wiley-Interscience, 2001.
- [6] D. Alspach and H. Sorenson, "Nonlinear Bayesian estimation using Gaussian sum approximations," *IEEE Transactions on Automatic Control*, vol. 17, no. 4, pp. 439–448, 1972.
- [7] H. Sorenson and D. Alspach, "Recursive Bayesian estimation using Gaussian sums," *Automatica*, vol. 7, no. 4, pp. 465–479, Jul. 1971.
- [8] M. Flament, G. Fleury, and M.-E. Davoust, "Particle filter and Gaussian-mixture filter efficiency evaluation for terrain-aided navigation," in *EUSIPCO 2004*, Sep. 2004.
- [9] T. Lehn-Schioler, D. Erdogmus, and J. Principe, "Parzen particle filters," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, vol. 5, 2004, pp. V–781–4 vol.5.

¹⁵If reduction is done by removing the components with smaller weights, as in (25)–(26), the two sides are equal.

TABLE VII
NUMBER OF REQUIRED OPERATIONS FOR UPDATE STAGE OF GSF

	Operation	additions	multiplications	Frequency
W_k^{ij}	$P_{k k-1}^{ij} H_k^T$	$n_x n_z (n_x - 1)$	$n_x^2 n_z$	filter
	W_k^{ij}	$n_x n_z (n_z - 1)$	$n_x n_z^2$	
$P_{k k}^{ij}$	$W_k^{ij} S_k^{ij}$	$n_x n_z (n_z - 1)$	$n_x n_z^2$	filter
	$W_k^{ij} S_k^{ij} W_k^{ijT}$	$n_x^2 (n_z - 1)$	$n_x^2 n_z$	
	$P_{k k-1}^{ij}$	n_x^2	0	
$\hat{x}_{k k}^{ij}$	$W_k^{ij} \nu_k^{ij}$	$n_x (n_z - 1)$	$n_x n_z$	filter
	$\hat{x}_{k k}^{ij}$	n_x	0	

TABLE VIII
NUMBER OF REQUIRED OPERATIONS FOR UPDATE STAGE OF APPROXIMATE MMSE FILTER

	Operation	additions	multiplications	Frequency
U_k		$C_{v_k} C_{w_k} - 1$	$n_x C_{v_k} C_{w_k}$	iteration
A_k^{ij}	$P_{k k-1}^{ij} H_k^T$	$n_x n_z (n_x - 1)$	$n_x^2 n_z$	filter
	$U_k - u_k^i$	n_x	0	
	$(U_k - u_k^i) \nu_k^{ijT}$	0	$n_x n_z$	
	$P_{k k-1}^{ij} H_k^T + (U_k - u_k^i) \nu_k^{ijT}$	$n_x n_z$	0	
	$S_k^{ij} + \nu_k^{ij} \nu_k^{ijT}$	n_z^2	n_z^2	
B_k^{ij}	A_k^{ij}	$n_x n_z (n_z - 1)$	$n_x n_z^2$	filter
		$n_z (n_z - 1)$	n_z^2	
S_k	$A_k^{ij} \nu_k^{ij}$	$C_{v_k} C_{w_k} n_x (n_z - 1)$	$C_{v_k} C_{w_k} n_x n_z$	iteration
	$\sum_{ij} \mu_k^{ij} A_k^{ij} \nu_k^{ij}$	$(C_{v_k} C_{w_k} - 1) n_x$	$C_{v_k} C_{w_k} n_x$	
	$B_k^{ij} \nu_k^{ij}$	$C_{v_k} C_{w_k} (n_z - 1)$	$C_{v_k} C_{w_k} n_z$	
	$\sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij}$	$C_{v_k} C_{w_k} - 1$	$C_{v_k} C_{w_k}$	
	$\left(1 - \sum_{ij} \mu_k^{ij} B_k^{ij} \nu_k^{ij}\right)^{-1}$	1	1	
S_k		0	n_x	
W_k^{ij}	$S_k B_k^{ij}$	0	$n_x n_z$	filter
	$A_k^{ij} + S_k B_k^{ij}$	$n_x n_z$	0	
$P_{k k}^{ij}$	$H_k P_{k k-1}^{ij}$	$n_x n_z (n_x - 1)$	$n_x^2 n_z$	filter
	$W_k^{ij} H_k P_{k k-1}^{ij}$	$n_x^2 (n_z - 1)$	$n_x^2 n_z$	
	$W_k^{ij} S_k^{ij}$	$n_x n_z (n_z - 1)$	$n_x n_z^2$	
	$W_k^{ij} S_k^{ij} W_k^{ijT}$	$n_x^2 (n_z - 1)$	$n_x^2 n_z$	
	$P_{k k}^{ij}$	$3n_x^2$	0	
$\hat{x}_{k k}^{ij}$	$W_k^{ij} \nu_k^{ij}$	$n_x (n_z - 1)$	$n_x n_z$	filter
	$\hat{x}_{k k}^{ij}$	n_x	0	

TABLE IX
NUMBER OF REQUIRED OPERATIONS FOR MERGING

	Operation	additions	multiplications	Frequency
$\hat{x}_{k k}$		$C_{v_k} C_{w_k} - 1$	$n_x C_{v_k} C_{w_k}$	iteration
$P_{k k}$	$(\hat{x}_{k k} - \hat{x}_{k k}^{ij})$	n_x	0	iteration
	$(\hat{x}_{k k} - \hat{x}_{k k}^{ij}) (\hat{x}_{k k} - \hat{x}_{k k}^{ij})^T$	0	n_x^2	
	$P_{k k}^{ij} + (\hat{x}_{k k} - \hat{x}_{k k}^{ij}) (\hat{x}_{k k} - \hat{x}_{k k}^{ij})^T$	n_x^2	0	
	$P_{k k}$	$C_{v_k} C_{w_k} - 1$	$n_x^2 C_{v_k} C_{w_k}$	

- [10] I. Bilik and J. Tabrikian, "Optimal recursive filtering using Gaussian mixture model," in *IEEE/SP 13th Workshop on Statistical Signal Processing*, 2005, pp. 399–404.
- [11] M. Huber and U. Hanebeck, "The hybrid density filter for nonlinear estimation based on hybrid conditional density approximation," in *10th International Conference on Information Fusion (FUSION)*, 2007, pp. 1–8.
- [12] M. Huber, D. Brunn, and U. Hanebeck, "Efficient nonlinear measurement updating based on Gaussian mixture approximation of conditional densities," in *American Control Conference (ACC)*, 2007, pp. 4425–

4430.

- [13] M. Huber and U. Hanebeck, "Hybrid transition density approximation for efficient recursive prediction of nonlinear dynamic systems," in *6th International Symposium on Information Processing in Sensor Networks (IPSN)*, 2007, pp. 283–292.
- [14] A. S. Stordal, H. A. Karlson, G. Nvdal, H. J. Skaug, and B. Valls, "Bridging the ensemble Kalman filter and particle filters: the adaptive Gaussian mixture filter," *Computational Geosciences*, vol. 15, no. 2, pp. 293–305, Mar. 2011.
- [15] S. Ali-Löytty, "Efficient Gaussian mixture filter for hybrid positioning,"

- in *IEEE/ION Position, Location and Navigation Symposium*, 2008, pp. 60–66.
- [16] J. Kotecha and P. Djuric, “Gaussian sum particle filtering,” *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2602–2612, Oct. 2003.
 - [17] R. Chen and J. S. Liu, “Mixture Kalman filters,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 3, pp. 493–508, 2000.
 - [18] J. R. Schoenberg, M. Campbell, and I. Miller, “Posterior representation with a multi-modal likelihood using the Gaussian sum filter for localization in a known map,” *Journal of Field Robotics*, vol. 29, no. 2, pp. 240–257, 2012.
 - [19] I. Bilik and J. Tabrikian, “MMSE-Based filtering in presence of non-Gaussian system and measurement noise,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 3, pp. 1153–1170, 2010.
 - [20] X. Sun, L. Munoz, and R. Horowitz, “Mixture Kalman filter based highway congestion mode and vehicle density estimator and its application,” in *American Control Conference*, vol. 3, 2004, pp. 2098–2103.
 - [21] S. Ali-Löytty, “Box Gaussian mixture filter,” *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2165–2169, 2010.
 - [22] F. Faubel, J. McDonough, and D. Klakow, “The split and merge unscented Gaussian mixture filter,” *IEEE Signal Processing Letters*, vol. 16, no. 9, pp. 786–789, 2009.
 - [23] L. Xiang and B. Su, “A modified particle filter for nonlinear systems with application to tracking problem,” in *7th World Congress on Intelligent Control and Automation (WCICA)*, 2008, pp. 4095–4099.
 - [24] Y. Wang, N. Dahnoun, and A. Achim, “A novel system for robust lane detection and tracking,” *Signal Processing*, vol. 92, no. 2, pp. 319–334, Feb. 2012.
 - [25] E. Daeipour and Y. Bar-Shalom, “An interacting multiple model approach for target tracking with glint noise,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 2, pp. 706–715, 1995.
 - [26] K. Ito and K. Xiong, “Gaussian filters for nonlinear filtering problems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 5, pp. 910–927, 2000.
 - [27] J. Kotecha and P. Djuric, “Gaussian particle filtering,” *IEEE Transactions on Signal Processing*, vol. 51, no. 10, pp. 2592–2601, Oct. 2003.
 - [28] D. Crisan and K. Li, “Generalised particle filters with Gaussian mixtures,” *arXiv preprint arXiv:1306.0255*, 2013.
 - [29] G. Terejanu, P. Singla, T. Singh, and P. Scott, “Adaptive Gaussian sum filter for nonlinear Bayesian estimation,” *IEEE Transactions on Automatic Control*, vol. 56, no. 9, pp. 2151–2156, 2011.
 - [30] P. Foo and G. Ng, “Combining the interacting multiple model method with particle filters for manoeuvring target tracking,” *IET Radar, Sonar Navigation*, vol. 5, no. 3, pp. 234–255, 2011.
 - [31] I. Bilik and J. Tabrikian, “Maneuvering target tracking in the presence of glint using the nonlinear Gaussian mixture Kalman filter,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, no. 1, pp. 246–262, 2010.
 - [32] O. C. Schrempf and U. D. Hanebeck, “Efficient recursive prediction of stochastic nonlinear systems based on Dirac mixture approximations,” in *In Proceedings of the American Control Conference*, 2007.
 - [33] R. E. Bethel, B. Shapo, and C. M. Kreucher, “PDF target detection and tracking,” *Signal Processing*, vol. 90, no. 7, pp. 2164–2176, Jul. 2010.
 - [34] M. West, “Approximating posterior distributions by mixture,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 55, no. 2, pp. 409–422, Jan. 1993.
 - [35] A. Runnalls, “Kullback-Leibler approach to Gaussian mixture reduction,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 43, no. 3, pp. 989–999, 2007.
 - [36] D. Schieferdecker and M. Huber, “Gaussian mixture reduction via clustering,” in *12th International Conference on Information Fusion (FUSION)*, Jul. 2009, pp. 1536–1543.
 - [37] J. Williams and P. Maybeck, “Cost-function-based Gaussian mixture reduction for target tracking,” in *6th International Conference of Information Fusion (FUSION)*, vol. 2, 2003, pp. 1047–1054.
 - [38] D. J. Salmond, “Mixture reduction algorithms for target tracking in clutter,” in *Proceedings of SPIE Signal and Data Processing of Small Targets*, vol. 1305, 1990, pp. 434–445.
 - [39] C. Andrieu and A. Doucet, “Particle filtering for partially observed Gaussian state space models,” *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 64, no. 4, p. 827836, 2002.
 - [40] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, “A look at Gaussian mixture reduction algorithms,” in *14th International Conference on Information Fusion (FUSION)*, Jul. 2011, pp. 1–8.
 - [41] H. V. Poor, *An introduction to signal detection and estimation*. New York, NY, USA: Springer-Verlag, 1994.
 - [42] G. Ackerson and K. Fu, “On state estimation in switching environments,” *IEEE Transactions on Automatic Control*, vol. 15, no. 1, pp. 10–17, 1970.
 - [43] J. Tugnait and A. Haddad, “Adaptive estimation in linear systems with unknown Markovian noise statistics,” *IEEE Transactions on Information Theory*, vol. 26, no. 1, pp. 66–78, 1980.